PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 17/00

A1

(11) International Publication Number: WO 99/66422

(43) International Publication Date: 23 December 1999 (23.12.99)

(21) International Application Number: PCT/US99/13488

(22) International Filing Date: 15 June 1999 (15.06.99)

(30) Priority Data:
60/089,212 15 June 1998 (15.06.98) US

(63) Related by Continuation (CON) or Continuation-in-Part (CIP) to Earlier Application
US 08/485,269 (CIP)
Filed on 7 June 1995 (07.06.95)

(71) Applicant *(for all designated States except US)*: ASCOM HASLER MAILING SYSTEMS, INC. [US/US]; 19 Forest Parkway, P.O. Box 858, Shelton, CT 06484-0904 (US).

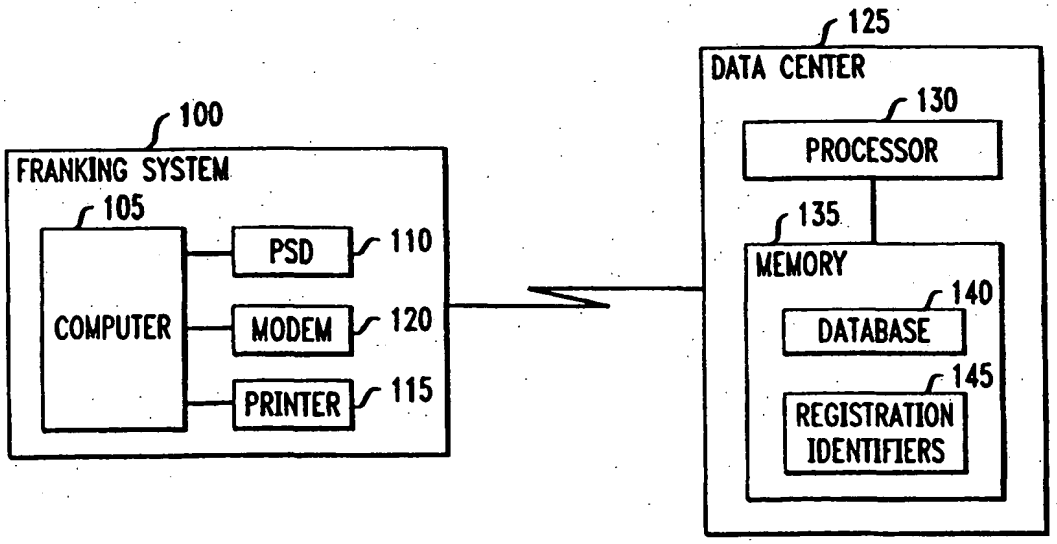
(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: SCHWARTZ, Robert, G. [US/US]; 191 Linden Avenue, Branford, CT 06405 (US). BROOKNER, George, M. [US/US]; 11 Surrey Drive, Norwalk, CT 06851 (US). ESKANDARI, Fetneh [IR/US]; 144 Dove Lane, Middletown, CT 06457 (US). CROWE, Allen, A. [US/US]; 76 Klein Drive, Prospect, CT 06712 (US). SIMCIK, Mark, E. [US/US]; 141 Park Avenue, Bloomfield, CT 06002 (US).

(74) Agent: YIP, Alex, L.; Londa & Traub LLP, 37th floor, 20 Exchange Place, New York, NY 10005 (US).

(81) Designated States: CA, JP, US, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.

(54) Title: TECHNIQUE FOR SECURING A SYSTEM CONFIGURATION OF A POSTAGE FRANKING SYSTEM



(57) Abstract

In a franking system a postal security device (PSD) tracks a postage fund for dispensing postal indicia and enforce the configuration of the franking system. An authorization code, which is particular to the system, is used to verify the system configuration. An unauthorized change in the system configuration causes invalidation of the code and generation of the postal indicia is denied. Data center (125) records configuration information of each franking system (100). The data center generates a valid authorization code for verification in the franking system based on new configuration information. Components added to the system must be preapproved to prevent fraudulent generation of postage indicia. A registration number is assigned to each preapproved component which is necessary for interaction with the franking system.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania
AM	Armenia
AT	Austria
AU	Australia
AZ	Azerbaijan
BA	Bosnia and Herzegovina
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CU	Cuba
CZ	Czech Republic
DE	Germany
DK	Denmark
EE	Estonia
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GH	Ghana
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IL	Israel
IS	Iceland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgyzstan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakstan
LC	Saint Lucia
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LS	Lesotho
LT	Lithuania
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
MK	The former Yugoslav Republic of Macedonia
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
MX	Mexico
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SG	Singapore
SI	Slovenia
SK	Slovakia
SN	Senegal
SZ	Swaziland
TD	Chad
TG	Togo
TJ	Tajikistan
TM	Turkmenistan
TR	Turkey
TT	Trinidad and Tobago
UA	Ukraine
UG	Uganda
US	United States of America
UZ	Uzbekistan
VN	Viet Nam
YU	Yugoslavia
ZW	Zimbabwe

Description

TECHNIQUE FOR SECURING A SYSTEM CONFIGURATION OF A POSTAGE FRANKING SYSTEM

Technical Field

The invention relates to a secure system configuration technique, and more particularly to a technique for protecting the integrity of components in a postage franking system.

Background of the Invention

It is commonplace to use postage meters or franking systems for generating postage indicia on mailpieces. The format of the postage indicia is specified by a postal authority to facilitate its inspection. In the United States, much attention has been focused on an Information-Based Indicia Program (IBIP) by the United States Postal Service (USPS), proposing, among other things, new requirements for the format of a postage indicium. Such new requirements were promulgated, e.g., in the "Information Based Indicia Program (IBIP) Open System Indicium Specification," dated August 19, 1998. For instance, the IBIP requires inclusion of a 2-dimensional (2-D) barcode in the postage indicium. Such a barcode represents postal information including postage, and a digital signature for authenticating the postal information, in accordance with a public key algorithm. One such public key algorithm may be the Digital Signature Algorithm (DSA) described,

e.g., in "Digital Signature Standard (DSS)," FIPS PUB 186, May 19, 1994.

In addition, under the IBIP, the requirements of a postal security device (PSD) supporting the creation of the postage indicium are specified, e.g., in the "Information Based Indicia Program (IBIP) Open System Postal Security Device (PSD) Specification," dated August 19, 1998. In accordance with the IBIP requirements, the PSD provides the aforementioned digital signature in the postage indicium, and dispenses and accounts for a postal fund stored therein in a secure manner.

With the advent of sophisticated and widely available general purpose computers, e.g., personal computers (PCs), it has become possible to use one such computer, by installing an appropriate postage generation program therein, to print postage indicia on a printer. Thus, a franking system may comprise a PC, and a PSD and printer serving as peripherals thereto, in accordance with an "open system" configuration. An advantage of adopting the open system configuration is that other mailing application software may also be installed by the user in the same PC to effectively generate mailpieces along with the postage indicia. For example, such mailing application software may include a billing program for charging postage back to different accounts, an envelope program for printing an address and a postage indicium on an envelope, an address cleansing program for correcting mailing addresses, etc.

However, the user of a franking system based on the open system configuration has full access to the hardware and software components in the system. As a result, these components including the aforementioned postage generation program are subject to tampering, and

fraudulent manipulation to generate unauthorized postage indicia.

Summary Of the Invention

In accordance with the invention, an authorization code is used to secure the configuration of a franking system. The authorization code is derived in part from system configuration information concerning, e.g., the enabled and disabled feature options, current version number of software, and the identity of a computer in the franking system (e.g., the serial number of the computer). Any unauthorized change in the system configuration results in an invalidation of the authorization code in the franking system, and denial of the franking operation. Thus, any system reconfiguration, e.g., a change in the feature options or software upgrade, must be effected using a new valid authorization code. Preferably, the authorization code verification is performed each time before the franking operation starts to forestall any fraudulent generation of postage indicia.

In accordance with an aspect of the invention, software code, e.g., the object code of a postage generation program, in the franking system is subject to error checking thereof. Thus, the above authorization code is also derived from error checking information, e.g., cyclic redundancy check (CRC) bits or checksum of the software code. Any tampering of the software also results in an invalidation of the authorization code.

In addition, to minimize the risk of fraudulent generation of postage indicia, franking-related software and hardware components by, e.g., third party vendors,

need to go through a pre-approval process before they are installed in the franking system to participate in the franking operation. For instance, in the pre-approval process, the components need to pass standardized tests to meet certain minimum requirements in, e.g., tamper resistance. In accordance with yet another aspect of the invention, a pre-approved software component is afforded a registration identifier which is necessary for the software component to participate in the franking operation. For example, the registration identifier needs to be produced for verification each time when the software component interacts with the aforementioned postage generation program. Similarly, a pre-approved hardware component is afforded a registration identifier which is necessary for its utility software to participate in the franking operation.

It is an object of the invention to control the configurations of the franking systems in the field. To that end, a data center keeps records of the latest configurations of the franking systems served by the data center, including the identities of the franking-related components in the respective systems. Such records can be used to control the configuration of each franking system. For example, with such records, the data center can generate the aforementioned authorization code for verification in each franking system to enforce its configuration.

It is another object of the invention to effectively conduct online transactions using postage funds. To that end, the aforementioned data center also keeps a customer account for replenishing a postage fund in each franking system. For example, software or a feature option for the franking system may be purchased

through a communication connection with the data center. Such an online transaction involves the data center's downloading the software to, or enabling the feature option of, the franking system through the communication connection, with the price of the software or feature option debited from its customer account in the data center.

Brief Description of the Drawing

Further objects, features and advantages of the invention will become apparent from the following detailed description taken in conjunction with the accompanying figures showing illustrative embodiments of the invention, in which:

Fig. 1 illustrates a franking system which is capable of communicating with a remote data center in accordance with the invention;

Fig. 2 illustrates the format of each record in a database in the remote data center;

Fig. 3 is a block diagram of a postal security device used in the franking system;

Fig. 4 is a flow chart depicting the steps of a postage generation program used in the franking system;

Fig. 5 illustrates a postage indicium generated by the postage generation program;

Fig. 6 illustrates an authorization code which needs to be verified in reconfiguring the franking system;

Fig. 7 is a flow chart depicting the steps taken by the franking system to verify the authorization code;

Figs. 8A and 8B jointly illustrate a process whereby the franking system can be remotely reconfigured through a communications connection;

Fig. 9 shows a variation of the design of the authorization code;

Fig. 10 illustrates a memory map of storage of feature option values;

Fig. 11 illustrates a process for generating the authorization code of Fig. 9 in changing a feature option in the franking system;

Fig. 12 illustrates a process for changing the feature option in the franking system using the authorization code of Fig. 9;

Fig. 13 illustrates a second process for changing the feature option in the franking system using the authorization code of Fig. 9;

Fig. 14 illustrates a memory map of storage of software version numbers;

Fig. 15 illustrates a process for updating a software version number in the franking system; and

Figs. 16A, 16B and 16C jointly illustrate a process for printing addresses and a postage indicium on an envelope using pre-approved components in the franking system.

Detailed Description

Fig. 1 illustrates franking system 100 embodying the principles of the invention for realizing mailing applications and generating postage indicia on mailpieces. In this particular illustrative embodiment, system 100 is configured as an open system, where computer 105 may be a conventional personal computer (PC)

serving as a host device, and where PSD 110, printer 115 and modem 120 are peripherals to computer 105. Alternatively, computer 105 may be a workstation or any other general purpose computing machine. Computer 105 may cause modem 120 to establish a communication connection through a communications network to, say, remote data center 125. Although modem 120 in this instance is shown as an external modem, it will be appreciated that any internal modem within computer 105 may be used, instead.

Data center 125 includes processor 130 which, among other things, maintains database 140 and registration identifiers 145 stored in memory 135 to serve different franking systems, e.g., franking system 100, communicates therewith to replenish their postage funds, and provides authorization codes to control their configurations in accordance with the invention.

Database 140 contains records concerning the respective franking systems served by data center 125. Fig. 2 illustrates the format of each record in database 140. In this instance, each franking system is identified by a PSD serial number in field 161 pre-assigned to its PSD. Field 163 contains account information such as a prefunded or credit escrow account balance for the franking system for conducting a telemeter setting (TMS) transaction described below. Field 165 includes configuration information (described below) concerning the configuration of the franking system to protect its integrity in accordance with the invention.

Fig. 3 illustrates PSD 110 which in this instance is realized as an integrated circuit (IC) module peripheral to computer 105. PSD 110 comprises secure

memory 200, processing unit 210 including one or more processors, and communications interface 220 (realizable as PCMCIA, serial or parallel interface) for interfacing with and insertion into a corresponding mating port (not shown) in computer 105.

Secure memory 200 is a nonvolatile memory which includes, among others, ascending register 230 and descending register 235. Ascending register 230 is used to keep track of an amount of postage dispensed. On the other hand, descending register 235 is used to keep track of the postage fund amount available for postage dispensation. When the value of descending register 235 decreases over time below a predetermined limit, computer 105 can no longer dispense postage until descending register 235 is reset. Such a reset may be achieved by way of electronic funds transfer, in accordance with a well-known TMS technique, via a communication connection (e.g., a dial-up connection or an Internet connection) to data center 125 through modem 120.

Using the TMS technique in this instance, the user need not carry PSD 110 to a postal authority for authorized resetting of descending register 235. To initiate a TMS process on computer 105, the user needs to meet certain access requirements. For example, the user may be required to enter a password, key, or biometric input (e.g., fingerprint) on computer 105 using an appropriate input device attached to computer 105. Verification of such an access entry ensures that the user is authorized to conduct such a process. After the access entry is verified, computer 105 initiates a call through modem 120 (alternatively via the Internet) to data center 125, requesting additional postage funds. Upon receipt of the call, processor 130 verifies in a

well known manner the current ascending and descending register values and other PSD data in secure memory 200 of PSD 110, and ascertains the availability of funds in the prefunded or credit escrow account of system 100. After the PSD data is validated and the account balance is found to be sufficient, processor 130 debits the account and remotely resets descending register 235 in PSD 110 accordingly.

System 100 in this instance may be used to generate postage indicia in accordance with the United States Postal Service (USPS) Information Based Indicia Program (IBIP) specification, namely, the "Information Based Indicia Program (IBIP) Open System Indicium Specification," dated August 19, 1998. To that end, secure memory 200 also includes a well-known digital signature algorithm (DSA) described, e.g., in "Digital Signature Standard (DSS)," FIPS PUB 186, May 19, 1994; and a private key and the corresponding public key in accordance with the DSA. The public key may be made available to the public in a PSD certificate in the postage indicia. For instance, using the DSA, unit 210 may sign specified postal data with an associated private key to generate a different digital signature to be included in each postage indicium. The postal authority then scans the postage indicium and verifies the digital signature to authenticate the postage indicium, in accordance with the DSA. It should be noted that instead of the DSA of the DSS, another well-known data authentication algorithm such as the RSA or Elliptic Curve algorithm may be used.

For postage franking operation, computer 105 is loaded with software components including postage generation program 300 for generating postage indicia.

Fig. 4 illustrates program 300 stored in a memory (not shown) in computer 105. Instructed by program 300, computer 105 prompts the user to enter mailing information concerning the destination zip code, weight, mail class (or rate category), any special services, etc., of a mailpiece to be mailed, as indicated at step 305. Assuming in this instance that a rate module is pre-installed in computer 105 which provides postage rate information, computer 105 at step 310 calculates the required postage based on the user entries and postage rate information. Otherwise, the user would be prompted to enter the required postage value for mailing the mailpiece. At step 313, computer 105 sends the data concerning the mail class and postage value to PSD 110. Instructed by a subroutine of program 300, unit 210 in PSD 110 deducts the required postage value from the available postal fund in descending register 235, and accordingly adds same to the dispensed fund in ascending register 230 to account for the transaction, as indicated at step 315. At step 317, unit 210 in accordance with the DSA of the DSS signs postal data concerning the mail class, postage value, ascending and descending register values, and date of mailing, together with other data pre-stored in memory 200 such as the software ID identifying program 300, device ID identifying PSD 110, and licensing zip code, resulting in a digital signature for authenticating the postage indicium to be generated. At step 320, computer 105 receives from PSD 110 the digital signature, ascending and descending register values, etc. At step 325, computer 105 prepares a print image of a postage indicium representing the required postal data and digital signature. Alternatively, unit 210 itself may create the print image of the postage

indicium and pass it onto computer 105. Upon receiving a print command, computer 105 transmits the print image to printer 115, which then prints the postage indicium on a label or an envelope fed to printer 115.

Fig. 5 illustrates one such postage indicium 500 which serves as proof of postage payment. Indicium 500 includes human readable portion 555 and machine readable portion 560. Portion 555 may include, e.g., the date of mailing, postage, device ID, originating town and zip code, mail class, etc. Machine readable portion 560, which is readable using an optical scanner, may include a 2-dimensional barcode representing data concerning the device ID, ascending and descending register values, postage value, digital signature, date of mailing, licensing zip code, software ID, PSD certificate, mail class, etc. Alternatively, machine readable portion 560 may comprise one or more data matrix symbols representing similar data, as described in PCT International Publication No. WO 99/16023, published on April 1, 1999.

Because of the open system configuration of franking system 100, the user has full access to hardware and software components in system 100. As a result, these components, e.g., postage generation program 300 described above, are subject to tampering and unauthorized use. In accordance with the invention, verification of an authorization code is required from time to time to prevent tampering and unauthorized use of the components of system 100.

Fig. 6 illustrates one such authorization code 600 used to prevent any tampering and unauthorized use of postage generation program 300 described above, and feature options available in system 100 which may include, e.g., a label printing option and other printer

options, a barcode scanner option, etc. System 100 is pre-loaded with software components necessary for providing these options. A valid authorization code, which is unique to system 100, needs to be entered onto system 100 in order to install or upgrade the code of program 300, and/or enable new feature options selected by the user. In response to a user request for a system reconfiguration involving the program code and/or feature options, authorization code 600 is generated by processor 130 in data center 125 and then provided either to the user via facsimile, email, telephone, etc., for the user to enter onto system 100 using, e.g., a keyboard attached to computer 105, or to system 100 directly via the aforementioned communication connection between data center 125 and system 100. As shown in Fig. 6, authorization code 600 consists of m-bit electronic signature 605 and n-bit encrypted option segment 610, where m and n are predetermined integers. To generate electronic signature 605, for example, a combination of (a) the identity of computer 105, which in this instance is the serial number of computer 105, (b) the hardware configuration identifier of computer 105, e.g., indicative of the type of processor and RAM capacity in computer 105, (c) the serial number of PSD 110, (d) the software version number of program 300, (e) error checking information, e.g., in this instance cyclic redundancy check (CRC) bits, resulting from performing a CRC on the code of program 300, and (f) an option number whose bit pattern corresponds to a particular combination of the enabled and disabled feature options for the postage franking operation. Item (c) is provided in field 161, and items (a), (b), and (d) through (f) are

provided in field 165 of the record pertaining to system 100 in database 140.

It should be noted at this point that item (e) in this instance is obtained by running a well known CRC algorithm, e.g., Reed Solomon CRC algorithm, on the object code of program 300 which is authorized in system 100. Alternatively, a checksum derived in a conventional manner from the object code may be used.

The derivation by processor 130 of electronic signature 605 involves encrypting the combination of items (a) through (f) in accordance with a first well known encryption algorithm. Signature 605 is then derived from the encrypted version of the combination of the items, e.g., by extracting therefrom a predetermined sequence of m bits. Alternatively, signature 605 may be generated using a well known symmetric or asymmetric key cryptographic methodology.

On the other hand, encrypted option segment 610 is generated by encrypting only the option number (f) in accordance with a second well known encryption algorithm. Alternatively, segment 610 may be unencrypted, i.e., containing the plain text of option number (f).

It suffices to know for now that after system 100 enters a reconfiguration mode where authorization code 600 is entered, code 600 is stored in authorization code buffer 241. Encrypted option segment 610 of code 600 is subsequently decrypted to recover the underlying option number. Using the recovered option number (f) and additional items in system 100 which are identical to aforementioned items (a) through (e), and the same first encryption algorithm in the above-described manner, system 100 is capable of independently generating an electronic signature identical to electronic signature

605 of code 600. In any event, the generated signature is compared with electronic signature 605 in buffer 241. If the two signatures match, the authorization code is declared valid. Otherwise, if they do not match, the franking operation by system 100 is suspended.

It should be noted at this point that the authorization code verification requirement is desirable in that it helps deter unauthorized copying of software in system 100, e.g., program 300, onto other similar systems. This stems from the fact that even though the software can be copied onto the similar systems, the latter would not be able to perform the franking operation without proper authorization codes, which need to be derived in part from their respective unique computer and PSD serial numbers. In addition, because authorization code 600 is partly derived from aforementioned item (e), tampering of the software is prevented as any such tampering results in a deviation from the valid CRC bit values, causing invalidation of the authorization code. Moreover, since system 100 would only be able to perform franking operation with a proper authorization code, which specifies a valid combination of software and hardware components, and feature options in system 100, the authorization code verification requirement thus enables data center 125 to control the configuration of each franking system served thereby.

As mentioned before, each bit of the option number (f) corresponds to a feature option of franking system 100. Each option, which is initially disabled, may be selectively enabled by setting the appropriate bits of the option number (f) to the opposite value. Thus, for example, if a user wants to enable a previously disabled label printing option, a proper authorization

code needs to be entered on system 100 while in a reconfiguration mode, causing the bit in the option number (f) corresponding to the label printing option to change to the opposite value to enable the option. System 100 effects the feature options according to the bit pattern of the option number stored in option number buffer 243 in memory 200. In this particular illustrative embodiment, the recovered option number from decrypting segment 610 of authorization code 600 overwrites the current option number in buffer 243 irrespective of the outcome of the validation of authorization number 600. That is, system 100 immediately effects the feature options according to the recovered option number as soon as it is placed in buffer 243, irrespective of the outcome of the validation.

After the feature options are effected in the prescribed manner in the reconfiguration mode, system 100 returns to a normal operation mode. When postage generation program 300 is invoked to perform the franking operation in the normal operation mode, unit 210 reads from memory 200 (i) the serial number of computer 105, (ii) the hardware configuration identifier of computer 105, (iii) the serial number of PSD 110, and (iv) the software version number of program 300, which are collected by unit 210 and stored in memory 200. Unit 210 also obtains (v) CRC bits based on running the aforementioned CRC algorithm on the latest code of program 300 in system 100, and (vi) the option number from buffer 243. Unit 210 independently generates an electronic signature using items (i) through (vi) and the aforementioned first encryption algorithm in a similar manner to processor 130 generating electronic signature 605 in data center 125. The electronic signature, thus

generated, is compared with the electronic signature stored in buffer 241, i.e., the first m bits of authorization code 600 therein. If there is no mismatch, generation of postage indicia using program 300 is allowed. Otherwise if there is any mismatch, a message such as "Invalid Authorization Code" is displayed on computer 105, and generation of postage indicia is halted.

Where authorization code 600 is entered by user onto system 100, in view of the possibility that the user makes an erroneous authorization code entry, the user is afforded a limited number of times to re-enter the correct authorization code after the message is displayed. After the limited number of times is exhausted, proper resetting of system 100 by authorized personnel is needed to re-enable the system to perform the franking operation.

For installing or upgrading a software component, e.g., the code of postage generation program 300, the user may be provided with a compact disk (CD), or another conventional storage medium, e.g., a floppy disk, IC module, digital video disk (DVD), etc., containing the necessary software, and authorization code 600 on the storage medium package which is generated in data center 125 for verification after the software installation or upgrade. The new software version number of program 300 may be embedded in the header of the program. When the software installation or upgrade is performed, the new software version number is read by computer 105 and transferred to memory 200 where the new software version number replaces the current software version number (iv).

After the software installation or upgrade in the reconfiguration mode, system 100 returns to the normal operation mode. When postage generation program 300 is invoked to perform the franking operation in the normal operation mode, the user is prompted for authorization code 600 on the storage medium package. Authorization code 600 is then verified according to the steps similar to those in the above-described verification after effecting new feature options. Specifically, unit 210 stores in buffer 241 authorization code 600 entered by the user, as indicated at step 701 in Fig. 7. At step 702, unit 210 causes the decryption of encrypted option segment 610 of authorization code 600 in buffer 241, thereby recovering the underlying option number (vi). Such decryption is accomplished using a decryption algorithm inverse to the second encryption algorithm. At step 703, processor 201 stores the recovered option number in buffer 243, although in this instance the recovered option number is identical to current option number in buffer 243. At step 704, unit 210 runs the CRC algorithm on the latest code of postage generation program 300, thereby obtaining item (v). At step 705, unit 210 reads the above items (i) through (iv) from memory 200, where item (iv) has the latest software version number of program 300. At step 706, unit 210 independently generates an electronic signature using items (i) through (vi), and the first encryption algorithm in a similar manner to processor 130 generating electronic signature 605 in data center 125. Unit 210 at step 707 compares the generated electronic signature with electronic signature 605 of authorization code 600 in buffer 241. The authorization code is validated if unit 210 finds that the two electronic signatures match.

Otherwise, a message such as "Invalid Authorization Code" is displayed on computer 105, and generation of postage indicia is halted.

It should be noted that the above authorization code verification is performed not only after system 100 is reconfigured, but preferably each time, or from time to time, when postage generation program 300 is invoked in the normal operation mode. Thus, preferably each time, or from time to time, before the franking operation is initiated, processor 201 performs above steps 702 through 707 for fear that the components of franking system 100 are tampered in the meantime.

It should also be noted that the above authorization code verification may also be performed via direct communications between data center 125 and franking system 100, thereby obviating the need of having the user enter the authorization code. Figs. 8A and 8B jointly illustrate remote reconfiguration process 800 whereby a user can purchase a new feature option or software online, and whereby authorization code 600 is verified via direct communications between data center 125 and system 100. Process 800 may be invoked by the user's entering a specified command on computer 105. Similar to the above-described TMS process for requesting additional postage, process 800 starts with prompting the user for an access entry (e.g., a password, key or biometric input) on computer 105, as indicated at step 806 in Fig. 8A. Verification of such an access entry ensures that the user is authorized to conduct the remote reconfiguration process. After the access entry is verified at step 809, computer 105 at step 812 establishes a communication connection with data center 125 via modem 120. Through the established connection,

processor 130 in data center 125 performs initial handshaking with franking system 100 according to a pre-agreed upon communication protocol, thereby identifying at step 815 franking system 100, e.g., by its PSD serial number. Based on the PSD serial number, processor 130 at step 818 locates in database 140 the record pertaining to franking system 100.

At step 821, processor 130 reviews fields 163 and 165 of the located record for the current escrow account balance and configuration information of system 100, respectively. Based on the current configuration of system 100, processor 130 at step 824 causes computer 105 to display a menu thereon containing selections of any new software available for downloading, and currently disabled options for activation. The menu also indicates the current escrow account or credit balance, the prices for downloading any new software having a new version number, and for activating one or more of the disabled options. Assuming that in this example the user wants to activate a previously disabled option, say, option A in the menu, the user may use a mouse device (not shown) attached to computer 105 to select option A.

At step 827, computer 105 communicates the user's selection of option A to processor 130. Upon receiving the option selection, processor 130 at step 830 debits the price of option A from the current escrow account balance, resulting in a new balance in field 163. Accordingly, processor 130 at step 833 changes the value of the bit in the option number (f) in field 165 corresponding to option A, reflecting an activation of option A. At step 836, processor 130 generates authorization code 600 consisting of electronic signature 605 and encrypted option segment 610. As mentioned

before, electronic signature 605 is derived from an encrypted version of items (a) through (f) in field 165 of the record pertaining to system 100. Encrypted option segment 610 is obtained by encrypting the option number (f) alone. Authorization code 600 is then transmitted from data center 125 to system 100 through the established communication connection, as indicated at step 839. The communication connection is thereafter terminated.

The remaining steps in process 800 are similar to those in routine 700 described before. Specifically, similar to step 701, step 841 in Fig. 8B involves storing received authorization code in buffer 241. Similar to step 702, step 843 involves decryption of encrypted option segment 610 of authorization code 600 to recover the underlying option number (vi), which in this instance indicates the activation status of option A. Similar to step 703, step 845 involves storing the recovered option number in buffer 243, thereby activating option A. Similar to step 704, step 847 involves running the CRC algorithm on the latest code of postage generation program 300, thereby obtaining item (v). Similar to step 705, step 849 involves reading items (i) through (iv) from memory 200. Similar to step 706, step 851 involves independently generating an electronic signature using items (i) through (vi), and the first encryption algorithm. Similar to step 707, step 853 involves comparing the generated electronic signature with electronic signature 605 of authorization code 600 in buffer 241. Again, the authorization code is validated if unit 210 finds that the two electronic signatures match. Otherwise, an "Invalid Authorization Code" message would be displayed on computer 105, and

generation of postage indicia would be halted as described before.

Based on the disclosure heretofore, it is apparent to a person skilled in the art that where the user chooses to purchase new software online, instead, the steps in process 800 similarly follow, except that in that case, at step 839 the new software, including the new software version number therein, would be downloaded from data center 125 to system 100, along with the transmission of authorization code 600 thereto.

Variations of the design of the authorization code which call for different verification techniques will now be described. In accordance with a first design variation, the authorization code is generated by encrypting items (a) through (f) using a standard encryption algorithm in data center 125. After such an authorization code is provided to system 100, the latter decrypts the received authorization code using a decryption algorithm inverse to the standard encryption algorithm, thereby recovering the underlying items (a) through (f). Items (i) through (v) are then obtain in system 100 in the manner described before, and compares them with the corresponding, recovered items (a) through (e). The authorization code is validated if the two sets of items match.

If the authorization code of the first design variation is not validated because of certain mismatched items, it may be desirable to show on computer 125 such mismatched items for diagnostic purposes. For example, if it is shown that item (d) does not match item (iv), a wrong software version of program 300 may have been installed in system 100. It may be a manufacturing

defect if the authorization code invalidation occurs during the very first time of the franking operation.

Fig. 9 illustrates a second variation of the authorization code design. In accordance with this variation, authorization code 900 includes m-bit electronic signature 905 which is generated in the same manner as electronic signature 605. Authorization code 900 also includes encrypted reconfiguration segment 910 having a variable length. The formation of segment 910 is fully described below. It suffices to know for now that the length of segment 910 depends on the actual reconfiguration which needs to be realized.

In a first example where authorization code 900 may be used, a user requests an activation of a currently disabled feature option, say, option C. In accordance with an aspect of the invention, for each feature option, a pair of memory locations are allocated in memory 200 of PSD 110 to pre-store "1" and "0" bit values representing, e.g., an "enabled" status and a "disabled" status of the option, respectively. The resulting memory map is illustrated in Fig. 10. As shown in Fig. 10, a first pair of memory addresses 1A2B (hexadecimal) and 1A2C in memory 200 correspond to feature option A, where "0" is pre-stored at memory address 1A2B and "1" is pre-stored at memory address 1A2C; a second pair of memory addresses 1A2D and 1A2E in memory 200 correspond to feature option B, where "0" is pre-stored at memory address 1A2D and "1" is pre-stored at memory address 1A2E; a third pair of memory addresses 1A2F and 1A30 in memory 200 correspond to feature option C, where "0" is pre-stored at memory address 1A2F and "1" is pre-stored at memory address 1A30; and so on and so forth. This memory map is made known to data center 125 beforehand and registered in

field 165 of the record pertaining to system 100 in database 140.

Continuing the above example, assuming that the request for activating feature option C is granted, processor 130 in data center 125 changes the value of the bit in option number (f) corresponding to option C from the previous value "0" to the new value "1" to activate the option, as indicated at step 1103 in Fig. 11. Processor 130 at step 1106 generates electronic signature 905 based on items (a) through (f) in the manner described before, where option number (f) incorporates the new bit value "1" corresponding to option C.

Processor 130 then generates encrypted reconfiguration segment 910. Specifically, at step 1109 processor 130 looks up from the aforementioned registered memory map the memory address corresponding to option C at which the new bit value "1" is pre-stored in memory 200. In this instance, the memory address in question is 1A30. At step 1112, processor 130 encrypts the memory address using the aforementioned second encryption algorithm, resulting in segment 910. Authorization code 900 consisting of electronic signature 905 and encrypted reconfiguration segment 910 is fed to system 100 in a reconfiguration mode either by direct communications or a user entry.

After receiving authorization code 900, unit 210 at step 1203 in Fig. 12 decrypts segment 910 of authorization code 900 using the decryption algorithm inverse to the second encryption algorithm, thereby recovering the memory address 1A30. It should be noted that segment 910 starts from the $(m+1)^{th}$ bit of received authorization code 900. Unit 210 at step 1206 retrieves from memory 200 the bit value "1" corresponding to option

C at memory address 1A30. Unit 210 at step 1209 overwrites the current bit value "0" corresponding to option C in option number buffer 243 with the retrieved bit value "1," thereby activating option C. Unit 210 at step 1212 gathers items (i) through (v) in the manner described before, and reads from option number buffer 243 the modified option number (vi). Unit 210 at step 1215 independently generates an electronic signature based on items (i) through (vi) in the manner described before. Unit 210 compares the resulting electronic signature with received electronic signature 905 of received authorization code 900, as indicated at step 1217. If they match, the authorization code is validated. Otherwise, an "Invalid Authorization Code" message would be displayed on computer 105, and generation of postage indicia would be halted as described before.

Although the above processes involve only one feature option, i.e., option C, the processes similarly follow where two or more options need to changed at the same time. In that case, the memory addresses associated with the multiple options are concatenated and then encrypted using the second encryption algorithm, thereby generating encrypted reconfiguration segment 910. Accordingly, the length of segment 910 increases with the number of feature options to be changed.

To keep segment 910 relatively short especially when multiple options need to be changed, in an alternative embodiment, segment 910 comprises an encrypted version of offset memory addresses, rather than full memory addresses, associated with the options. Referring briefly to Fig. 10, since the full memory address associated with each feature option illustratively starts with "1A," unit 210 can be

programmed to assume that the first two nibbles of the option memory addresses are always "1A". Thus, when option A needs to be changed, only the offset address "2B" or "2C" needs to be communicated using segment 910 for enabling or disabling the option; when option B needs to be changed, only the offset address "2D" or "2E" needs to be communicated using segment 910 for enabling or disabling the option; when option C needs to be changed, only the offset address "2F" or "30" needs to be communicated using segment 910 for enabling or disabling the option; and so on and so forth.

In a second example where authorization code 900 may be used, to save memory space in memory 200, the storage of "1" and "0" values for each option as set forth in the memory map of Fig. 10 may be totally avoided. Since a change in each option involves changing the corresponding bit value in option number buffer 243 to the opposite value, the encrypted reconfiguration segment 910 only needs to communicate the identities of the feature options which need to be changed. After learning the identities of such options based on segment 910, unit 210 locate the bits in buffer 243 corresponding to the identified options and change their current bit values to the opposite values, respectively.

Thus, in this second example, segment 910 is formed by encrypting codes identifying the respective options to be changed. Various designs of the codes are possible as long as each code uniquely identifies a respective option. For example, for the sake of convenience, the code identifying an option may represent the bit position corresponding to the option in buffer 243. Thus, the code for option A may be "01" representing the first bit position of buffer 243

corresponding to option A; the code for option B may be "02" representing the second bit position of buffer 243 corresponding to option B; the code for option C may be "03" representing the third bit position of buffer 243 corresponding to option C; and so on and so forth.

Continuing the second example, let's say that feature options A and C need to be changed in this instance. Thus, system 100 is fed with authorization code 900 wherein electronic signature 905 is generated by processor 130 in data center 125 in the manner described before, and encrypted reconfiguration segment 910 contains an encrypted version of the option codes "0103" in concatenation, where the option code "01" identifies option A and option code "03" identifies option C.

As indicated at step 1303 in Fig. 13, unit 210 first decrypts encrypted reconfiguration segment 910 of received authorization code 900, thereby recovering the option codes "0103". Based on a first option code "01" representing the first bit position in buffer 243 corresponding to option A, which needs to be changed, unit 210 at step 1306 changes the current value of the first bit in buffer 243 to the opposite value. In addition, based on a second option code "03" which immediately follows "01" and which represents the third bit position in buffer 243 corresponding to option C, which needs to be changed, unit 210 at step 1309 changes the current value of the third bit in buffer 243 to the opposite value. Unit 210 at step 1312, similar to above-described step 1212, gathers items (i) through (v), and reads from option number buffer 243 the modified option number (vi). Unit 210 at step 1315, similar to above-described step 1215, independently generates an electronic signature based on items (i) through (vi).

Unit 210 compares the resulting electronic signature with electronic signature 905 of received authorization code 900, as indicated at step 1317 similar to above-described step 1217. If they match, the authorization code is validated. Otherwise, an "Invalid Authorization Code" message would be displayed on computer 105, and generation of postage indicia would be halted as described before.

We have recognized that for loading new software on system 100 for a program upgrade or installation without changing feature options, authorization code 900 may consist of electronic signature 905 only, i.e., encrypted reconfiguration segment having a zero length. In this illustrative embodiment, an array of memory addresses in memory 200 are allocated to pre-store a quantity of possible version numbers of software, e.g., postage franking program 300. As shown in Fig. 14, for example, version number "1" is pre-stored at memory address 1B12; version number "2" is pre-stored at memory address 1B13; version number "3 is pre-stored at memory address 1B14; and so on and so forth. A version number pointer (not shown) in memory 200 is used to indicate the memory location of the current software version number. Assuming that the current software version number is "2", the pointer has a value of "1B13".

The new software to be loaded onto system 100 contains a header which in this instance includes the memory address at which the new software version number is pre-stored. Let's say the new version number is "3" and the header thus contains the memory address "1B14".

In granting the loading of new software onto system 100, processor 130 in data center 125 generates

authorization code 900 consisting of only electronic signature 905 based on items (a) through (f) in the manner described before, where item (d) has the new software version number. Electronic signature 905 is provided to system 100 for later verification.

While the new software is being loaded onto system 100 via an online connection or a storage medium, unit 210 in PSD 110 at step 1503 in Fig. 15 changes the aforementioned version number pointer value to the memory address provided in the header of the new software, i.e., "1B14". As a result, the pointer indicates a new memory location containing the software version number "3". Unit 210 at step 1506 gathers items (i) through (iii), (v) and (vi), and reads from memory address 1B14 indicated by the pointer the new software version number "3" as item (iv). Unit 210 at step 1509, similar to above-described step 1215, independently generates an electronic signature based on items (i) through (vi). Unit 210 compares the resulting electronic signature with received electronic signature 905, as indicated at step 1511 similar to above-described step 1217. If they match, the authorization code is validated. Otherwise, an "Invalid Authorization Code" message would be displayed on computer 105, and generation of postage indicia would be halted as described before.

It should be noted at this point that the memory address communicated in the header of the new software may be an offset address, as well, e.g., "12", "13", "14" . . ., rather than its full address, e.g., "1B12", "1B13", "1B14" . . . as it is understood that the two most significant nibbles of the full address are always "1B".

In addition, to save memory space in memory 200, the storage of possible software version numbers as set forth in the memory map of Fig. 14 may be totally avoided, especially where the software version number always increments by one when new software is loaded onto system 100. In that case, a counter (not shown) in PSD 110 may be used to keep track of the current software version number. Unit 210 may be programmed to be responsive to loading of new software onto system 100 to cause the counter to increment by one, thereby updating the software version number (iv). After loading of the new software, unit 210 independently generates an electronic signature based on items (i) through (vi). The generated electronic signature is compared with electronic signature 905 generated by data center 125 in part based on the new software version number in (d). If they match, the loading of new software onto system 100 is authorized.

Because system 100 is configured as an open system, a user may freely load additional software onto computer 105, and add to system 100 hardware components, e.g., peripherals to computer 105. An advantage of adopting the open system configuration is that application software, other than postage generation program 300 described above, may be installed by the user on his/her own in computer 105 to interact with, say, program 300, to realize a more comprehensive mailing process. Such other application software may include, e.g., a billing program for charging postage back to different accounts, an envelope program for printing an address and a postage indicium on an envelope, an address cleansing program for correcting mailing addresses, etc.

On the other hand, because system 100 is configured as an open system, the integrity of the franking operation thereby may be jeopardized. For example, the user may load illegitimate software on computer 105 to interact with postage generation program 300 to fraudulently print postage indicia. The user may also employ a printer of inferior quality to print substandard postage indicia, which are unreadable by an optical scanner.

Thus, in accordance with an aspect of the invention, the franking-related hardware and software components in system 100 need to be pre-approved. To that end, the components by different vendors need to pass standardized tests to meet certain minimum requirements in, e.g., compatibility with a postage generation program in the franking system, print quality, tamper resistance, efficiency, durability, etc., to become approved. The pre-approved components may then be marketed to users for installation in their franking systems, e.g., system 100. The manner in which the pre-approval requirement of the software and hardware components is enforced when they interact with the postage generation program is fully described below. It suffices to know for now that each pre-approved software component includes a valid registration identifier which is necessary for the software component to interact with the postage generation program. Similarly, for each pre-approved hardware component (e.g., a printer), its utility software (e.g., printer driver software) interfacing the hardware component with the postage generation program also includes a valid registration identifier, which is necessary for it to interact with the postage generation program.

In accordance with another aspect of the invention, a registration identifier is used to (1) identify a franking-related hardware or software component in a franking system configuration, (2) enforce the pre-approval requirement of such a hardware or software component. To achieve object (1), each pre-approved software component, and hardware component including its utility software is assigned a different registration identifier. A duplicate copy of the registration identifier is registered in memory 135 of data center 125. Thus, data center 125 includes in memory 135 a collection of registration identifiers 145 which identify and are associated with different pre-approved components. The registration identifier collection is updated from time to time as additional software and hardware component pass the standardized tests and become approved.

When each pre-approved component interacts with the postage generation program, the registration identifier in the component is compared with the registered registration identifier. If the two identifiers match or correspond, the component is verified to be pre-approved, thereby achieving object (2).

A pre-approved envelope program having a registration identifier for verification of its pre-approval status will now be described. This envelope program may be purchased from a third-party vendor and installed by the user in computer 105. Because of its pre-approval status, the envelope program includes therein a registration identifier which identifies the program. Figs. 16A, 16B and 16C jointly illustrate the envelope program and interactions with postage generation

program 300 to print addresses and a postage indicium on an envelope. Instructed by such an envelope program, computer 105 elicits from the user the size of the envelope to be used for a mailpiece, as indicated at step 1603 in Fig. 16A. Computer 105 at step 1606 displays an image of the envelope having the specified size on its screen. Computer 105 at step 1609 prompts the user to type originating mailing address and destination mailing address at desired locations on the displayed envelope. Computer 105 at step 1612 prompts the user to indicate the desired location on the displayed envelope where a postage indicium is to be printed. Accordingly, the user utilizes a mouse device to indicate the desired location which, in this instance, is the upper right corner of the envelope according to the postal authority regulations.

Computer 105 thereafter provides at step 1615 a draft option which enables the user to preview the envelope including a specimen indicium appearing at the user defined location before the envelope is printed. Thus, this option allows the user to check the format of the envelope and the relative placement of the address blocks, and postage indicium on the envelope before the user is committed thereto.

After the user decides to proceed with the printing of the envelope at step 1617, computer 105 at step 1618 generates a first ensemble of control characters indicating the position of the originating mailing address, a second ensemble of control characters indicating the position of the destination mailing address, and a third ensemble of control characters indicating the position of the postage indicium on the envelope. At step 1621, computer 105 inserts the first, second and third ensembles of control characters into the

data stream representative of the texts of the originating and destination mailing addresses, where the originating mailing address data is preceded by the first ensemble of control characters, and the destination mailing address data is preceded by the second ensemble of control characters. The resulting data stream is formatted pursuant to the protocol required by printer 115. For example, if printer 115 is a printer manufactured by Hewlett-Packard Co., the data stream would be in accordance with the Hewlett-Packard printer control language (HP-PCL).

The envelope program proceeds from step 1621 to step 1623 in Fig. 16B where postage generation program 300 described before is invoked. Upon such an invocation, unit 210 in PSD 110 is interrupted, and requests computer 105 to pass thereto a copy of the registration identifier in the envelope program for examination, as indicated at step 1624. If computer 105 fails to produce a copy of the registration identifier, unit 210 causes computer 105 to display thereon an "Unauthorized Component" message, and prevents generation of any postage indicium, as indicated at step 1625.

Otherwise, if computer 105 produces a copy of the registration identifier of the envelope program, unit 210 at step 1626 compares the registration identifier from computer 105 with each of registration identifiers 245 in PSD 110, which are associated with the pre-approved components which have been verified at least once. At step 1627, unit 210 determines whether a corresponding registration identifier is found amongst registration identifiers 245. Assuming that this is not the first time that the envelope program invokes program 300, and the registration identifier of the envelope

program has been verified at least once, unit 210 in this instance finds the corresponding registration identifier amongst registration identifiers 245, and proceeds to step 1642 in Fig. 16C described below.

Otherwise, if the registration identifier of the envelope program has never been verified, unit 210 fails to find a corresponding registration identifier amongst registration identifiers 245. Unit 210 then causes modem 120 to establish at step 1628 a communication connection with data center 125. Unit 210 transmits at step 1629 the serial number of PSD 110 and copy of the registration identifier of the envelope program to data center 125 where processor 130 at step 1630 compares the received registration identifier with each of registration identifiers 145 in data center 125, which as mentioned before consist of the registration identifiers of all pre-approved components ever. Processor 130 at step 1631 determines whether a corresponding registration identifier is found amongst registration identifiers 145.

Since in this instance, the envelope program is pre-approved, processor 130 locates a corresponding registration identifier amongst registration identifiers 145. Processor 130 recognizes that the envelope program identified by the corresponding registration identifier is being run on system 100, which is identified by the received serial number of PSD 110. Accordingly, processor 130 at step 1633 updates the record of system 100 in database 140 to also include in field 165 thereof an indication that the envelope program is now part of the configuration of system 100. Processor 130 then at step 1636 returns the copy of the registration identifier of the envelope program to unit 210, with an

acknowledgment that such a registration identifier is valid, and then terminates the communication connection. In response, unit 210 at step 1639 in Fig. 16C adds the returned registration identifier to registration identifiers 245 in PSD 110 for subsequent verification, obviating the need to have processor 130 involved in the subsequent verification of such a registration identifier. Unit 210 then goes on to help generate a postage indicium, as indicated at step 1642.

Otherwise, if processor 130 at step 1631 fails to locate a corresponding registration identifier amongst registration identifiers 145, processor 130 at step 1645 in Fig. 16B returns only a negative acknowledgement that the received registration identifier is invalid, and terminates the communication connection. In response to the negative acknowledgement, unit 210 returns to step 1625.

After step 1642 in Fig. 16C and execution of program 300, a print image of an appropriate postage indicium is prepared. At step 1648 a printer driver program associated with printer 115 is invoked to print the originating and destination addresses, and postage indicium on an envelope fed to printer 115. As the printer driver program interacts with program 300 to receive the print image of the postage indicium resulting from program 300, printer 115 including the printer driver program needs to be pre-approved. As such, upon the invocation of the printer driver program, unit 210 in PSD 110 is interrupted, and requests computer 105 to pass thereto a copy of the registration identifier in the printer driver program for examination, as indicated at step 1651. If computer 105 fails to produce a copy of such a registration identifier, unit 210 denies the

printer driver program of the print image of the postage indicium, as indicated at step 1654.

Otherwise, if computer 105 produces a copy of the registration identifier, unit 210 at step 1657 compares the registration identifier from computer 105 with each of registration identifiers 245 in PSD 110 which, as mentioned before, are associated with the pre-approved components which have been verified at least once. Assuming that this is not the first time that the printer driver program is invoked to print a postage indicium, and the registration identifier of the printer driver program has been verified at least once, unit 210 in this instance locates at step 1660 the corresponding registration identifier amongst registration identifiers 245. The printer driver program is provided with the print image of the postage indicium, as indicated at step 1663. At step 1667, printer 115 prints on the provided envelope the originating and destination addresses and the postage indicium at the user defined positions, based on the aforementioned data stream from computer 105 and the print image of the postage indicium.

Otherwise, if at step 1660 unit 210 fails to locate the corresponding registration identifier, processor 130 would be involved in verifying the registration identifier with the steps similar to steps 1628 through 1631, and 1633, 1636, 1639 and 1645 described before, which are not repeated here.

It is apparent from the disclosure heretofore that database 140 in data center 125 has records of configurations of all of the franking systems served by center 125. In particular, field 165 of each record pertaining to a respective franking system includes configuration information concerning, among others, the

hardware configuration of the computer (i.e., item (b)), the enabled or disabled options (i.e., item (f)), the version of the postage generation program (i.e., item (d)), and other hardware and software components interacting with the postage generation program in the franking system. Such information in database 140 can be used by a postal authority to effectively monitor and control the configurations of individual franking systems in the field.

The foregoing merely illustrates the principles of the invention. It will thus be appreciated that those skilled in the art will be able to devise numerous other arrangements which embody the principles of the invention and are thus within its spirit and scope.

For example, to further deter unauthorized reconfiguration of system 100, the encryption algorithms for generating authorization codes may be changed from time to time. The new algorithms may easily be downloaded from data center 125 during a software upgrade in computer 105, or during a TMS transaction with data center 125. The memory locations in the memory maps of Figs. 10 and 14 may be changed from time to time, as well.

In addition, in the illustrative embodiment, the memory of computer 105 is distinguished from memory 200 in PSD 110. However, the memory spaces in the two memories may be interchangeable in that some or all of the memory contents in memory 200 may be stored in the memory of computer 105, and vice versa. Similarly, some or all of the tasks performed by processing unit 210 in PSD 110 in the illustrative embodiment may be performed by computer 105, and vice versa.

Finally, the illustrative embodiment of the invention is disclosed herein in a form in which various franking and communications functions are performed by discrete functional blocks. These functional blocks may be implemented in various ways and combinations using logic circuitry and/or appropriately programmed processors, as will be known to those skilled in the art.

Claims

1. A franking system comprising:

a memory for storing a software component for generating at least one postage indicium;

a device for receiving an authorization code which is derived from at least information concerning the software component; and

a processing unit for verifying at least part of the authorization code to detect any change in the software component before the at least one postage indicium is generated.

2. The system of claim 1 wherein the information represents a version number of the software component.

3. The system of claim 2 further comprising a counter for keeping track of the version number of the software component.

4. The system of claim 2 wherein memory locations are allocated in the memory for storing a plurality of version numbers of the software component, respectively, the version number of the software component being indicated as stored at one of the memory locations.

5. The system of claim 1 wherein the information is obtained from running a predetermined algorithm on code of the software component.

6. The system of claim 5 wherein the information includes error checking information.

7. The system of claim 6 wherein the error checking information includes cyclic redundancy check (CRC) bits.

8. The system of claim 6 wherein the error checking information includes a checksum.

9. The system of claim 1 further comprising a computer where the memory is in, wherein the authorization code is also derived from an identity of the computer.

10. The system of claim 9 wherein the identity of the computer includes a serial number thereof.

11. The system of claim 1 further comprising a postal security device (PSD) where the processing unit is in, wherein the authorization code is also derived from an identity of the PSD.

12. The system of claim 11 wherein the identity of the PSD includes a serial number thereof.

13. A franking system comprising:

a memory for storing a software component for generating at least one postage indicium;

a buffer for storing an authorization code which is derived from at least information concerning a configuration of the system; and

a processing unit for verifying at least part of the authorization code before the at least one postage indicium is generated to detect any change in the configuration of the franking system.

14. The system of claim 13 further comprising software components for providing feature options in the system which are selectively enabled, wherein the configuration concerns at least a setting of the feature options.

15. The system of claim 13 wherein the configuration concerns at least a version of the software component.

16. The system of claim 13 further comprising a device for maintaining a postage fund for postage dispensation in the system, wherein the processing unit is within the device.

17. The system of claim 16 wherein the authorization code is also derived from an identity of the device.

18. The system of claim 17 wherein the identity of the device includes a serial number thereof.

19. The system of claim 13 further comprising a computer where the memory is in, wherein the authorization code is also derived from an identity of the computer.

20. The system of claim 19 wherein the identity of the computer includes a serial number thereof.

21. A franking system for generation of postage indicia, the system having a plurality of feature options which may be enabled, the system comprising:

a device for receiving an authorization code which is generated outside the system in response to a request for a selected setting of the feature options different from a current setting thereof, the authorization code

comprising a code segment and a data segment, the code segment being derived from at least information concerning the selected setting of the feature options, the data segment containing data concerning one or more of the feature options;

a buffer for effecting the selected setting of the feature options based on the data; and

a processing unit for verifying the code segment to determine whether generation of postage indicia based on the selected setting of the feature options is allowed.

22. The system of claim 21 wherein the data includes the information concerning the setting of the feature options.

23. The system of claim 21 wherein the data is encrypted.

24. The system of claim 21 wherein the selected setting of the feature options involves changing one or more of the feature options, with respect to the current setting of the feature options, the length of the data segment being a function of a quantity of the one or more of the feature options.

25. The system of claim 24 wherein the data indicates memory addresses which are associated with the one or more of the feature options, respectively, a value being stored at each memory address and the feature option associated with the memory address is changed to the value.

26. The system of claim 25 wherein the data includes offset memory addresses which are associated with the one or more of the feature options, respectively.

27. The system of claim 24 wherein the data identifies the one or more of the feature options.

28. A franking system comprising:

a first memory for storing a first software component for realizing at least one postage indicium, a second software component being stored in the first memory for interacting with the first software component, the second software component including a selected identifier;

a second memory for storing a plurality of identifiers; and

a processing unit for determining whether one of the plurality of identifiers corresponds to the selected identifier in the second software component when the second software component interacts with the first software component, the at least one postage indicium being realized only when one of the plurality of identifiers corresponds to the selected identifier.

29. The system of claim 28 further comprising a device for maintaining a postage fund for postage dispensation in the system, wherein the second memory is within the device.

30. The system of claim 28 wherein the selected identifier identifies the second software component.

31. The system of claim 28 further comprising at least one hardware component, wherein the second software component includes utility software for interfacing the first software component with the at least one hardware component.

32. A system for reconfiguring a franking apparatus for generating postage indicia, the franking apparatus including a device for maintaining a postage fund for postage dispensation in the franking apparatus, the system comprising:

a memory for storing a value of an account for replenishing the postage fund in the franking apparatus; and

a processor for reconfiguring the franking apparatus, a reconfiguration of the franking apparatus incurring a cost, the value of the account being adjusted to account for the cost, the value of the postage fund in the franking apparatus being unaffected by the reconfiguration.

33. The system of claim 32 wherein the franking apparatus is remotely reconfigured through a communication connection.

34. The system of claim 32 wherein the reconfiguration of the franking apparatus concerns at least a setting of feature options in the franking apparatus.

35. The system of claim 32 wherein the reconfiguration of the franking apparatus concerns at least a version of a software component in the franking apparatus.

36. The system of claim 32 wherein the memory also stores information concerning a current configuration of the franking apparatus.

37. The system of claim 36 wherein the processor causes transmission of a menu to the franking apparatus for the reconfiguration thereof, the menu being generated based on the information.

38. A method for use in a franking system comprising:

storing a software component for generating at least one postage indicium;

receiving an authorization code which is derived from at least information concerning the software component; and

verifying at least part of the authorization code to detect any change in the software component before the at least one postage indicium is generated.

39. The method of claim 38 wherein the information represents a version number of the software component.

40. The method of claim 39 further comprising keeping track of the version number of the software component using a counter in the system.

41. The method of claim 39 further comprising allocating memory locations to store a plurality of version numbers of the software component, respectively, the version number of the software component being indicated as stored at one of the memory locations.

42. The method of claim 38 wherein the information is obtained from running a predetermined algorithm on code of the software component.

43. The method of claim 42 wherein the information includes error checking information.

44. The method of claim 43 wherein the error checking information includes CRC bits.

45. The method of claim 43 wherein the error checking information includes a checksum.

46. The method of claim 38 wherein the authorization code is also derived from an identity of a computer in the system.

47. The method of claim 46 wherein the identity of the computer includes a serial number thereof.

48. The method of claim 38 wherein the authorization code is also derived from an identity of a PSD in the system.

49. The method of claim 38 wherein the identity of the PSD includes a serial number thereof.

50. A method for use in a franking system comprising:

storing a software component for generating at least one postage indicium;

storing an authorization code which is derived from at least information concerning a configuration of the system; and

verifying at least part of the authorization code before the at least one postage indicium is generated to detect any change in the configuration of the franking system.

51. The method of claim 50 further comprising providing feature options in the system which are selectively enabled, wherein the configuration concerns at least a setting of the feature options.

52. The method of claim 50 wherein the configuration concerns at least a version of the software component.

53. The method of claim 50 wherein the authorization code is also derived from an identity of a device for maintaining a postage fund for postage dispensation in the system.

54. The method of claim 53 wherein the identity of the device includes a serial number thereof.

55. The method of claim 50 wherein the authorization code is also derived from an identity of a computer.

56. The method of claim 55 wherein the identity of the computer includes a serial number thereof.

57. A method for use in a franking system for generation of postage indicia, the system having a plurality of feature options which may be enabled, the method comprising:

receiving an authorization code which is generated outside the system in response to a request for a

selected setting of the feature options different from a current setting thereof, the authorization code comprising a code segment and a data segment, the code segment being derived from at least information concerning the selected setting of the feature options, the data segment containing data concerning one or more of the feature options;

effecting the selected setting of the feature options based on the data; and

verifying the code segment to determine whether generation of postage indicia based on the selected setting of the feature options is allowed.

58. The method of claim 57 wherein the data includes the information concerning the setting of the feature options.

59. The method of claim 57 wherein the data is encrypted.

60. The method of claim 57 wherein the selected setting of the feature options involves changing one or more of the feature options, with respect to the current setting of the feature options, the length of the data segment being a function of a quantity of the one or more of the feature options.

61. The method of claim 60 wherein the data indicates memory addresses which are associated with the one or more of the feature options, respectively, a value being stored at each memory address and the feature option associated with the memory address is changed to the value.

62. The method of claim 61 wherein the data includes offset memory addresses which are associated with the one or more of the feature options, respectively.

63. The method of claim 57 wherein the data identifies the one or more of the feature options.

64. A method for use in a franking system comprising:

storing a first software component for realizing at least one postage indicium;

storing a second software component for interacting with the first software component, the second software component including a selected identifier;

storing a plurality of identifiers;

determining whether one of the plurality of identifiers corresponds to the selected identifier in the second software component when the second software component interacts with the first software component; and

realizing the at least one postage indicium when one of the plurality of identifiers corresponds to the selected identifier.

65. The method of claim 64 wherein the selected key identifies the second software component.

66. The method of claim 64 wherein the second software component includes utility software for interfacing the first software component with at least one hardware component in the system.

67. A method for reconfiguring a franking apparatus for generating postage indicia, the franking apparatus including a device for maintaining a postage fund for postage dispensation in the franking apparatus, the method comprising:

storing a value of an account for replenishing the postage fund in the franking apparatus;

reconfiguring the franking apparatus, a reconfiguration of the franking apparatus incurring a cost; and

adjusting the value of the account to account for the cost, the value of the postage fund in the franking apparatus being unaffected by the reconfiguration.

68. The method of claim 67 wherein the franking apparatus is remotely reconfigured through a communication connection.

69. The method of claim 67 wherein the reconfiguration of the franking apparatus concerns at least a setting of feature options in the franking apparatus.

70. The method of claim 67 wherein the reconfiguration of the franking apparatus concerns at least a version of a software component in the franking apparatus.

71. The method of claim 67 further comprising storing information concerning a current configuration of the franking apparatus.

72. The method of claim 71 further comprising transmitting a menu to the franking apparatus for the

reconfiguration thereof, the menu being generated based on the information.

FIG. 1

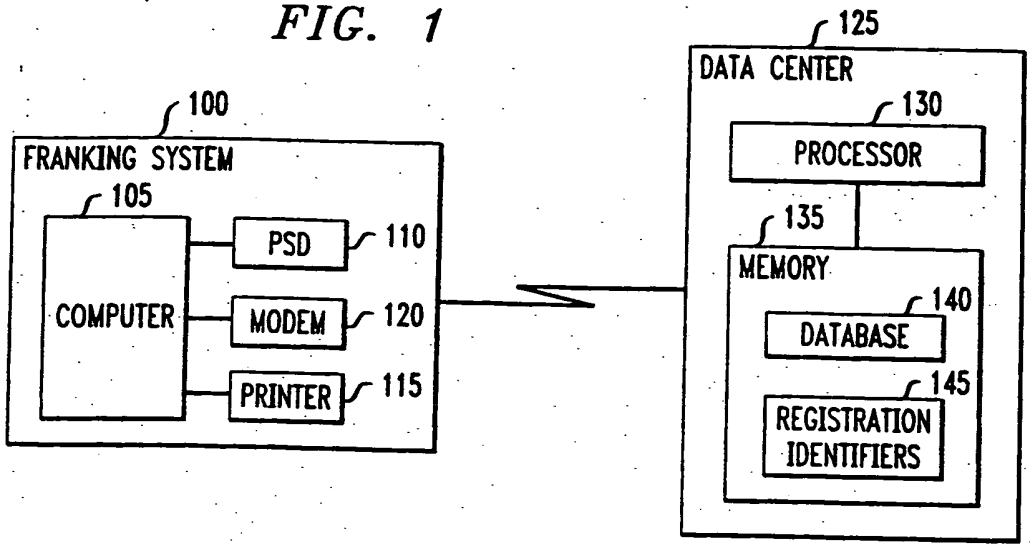


FIG. 2

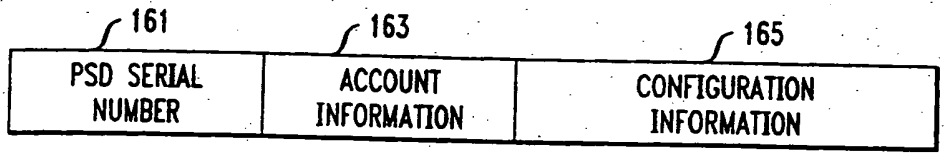


FIG. 3

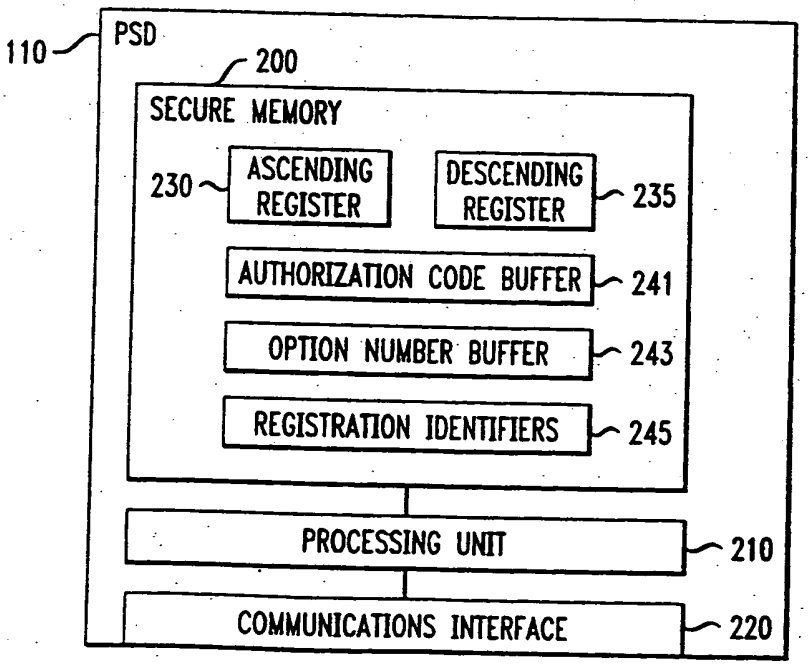


FIG. 4

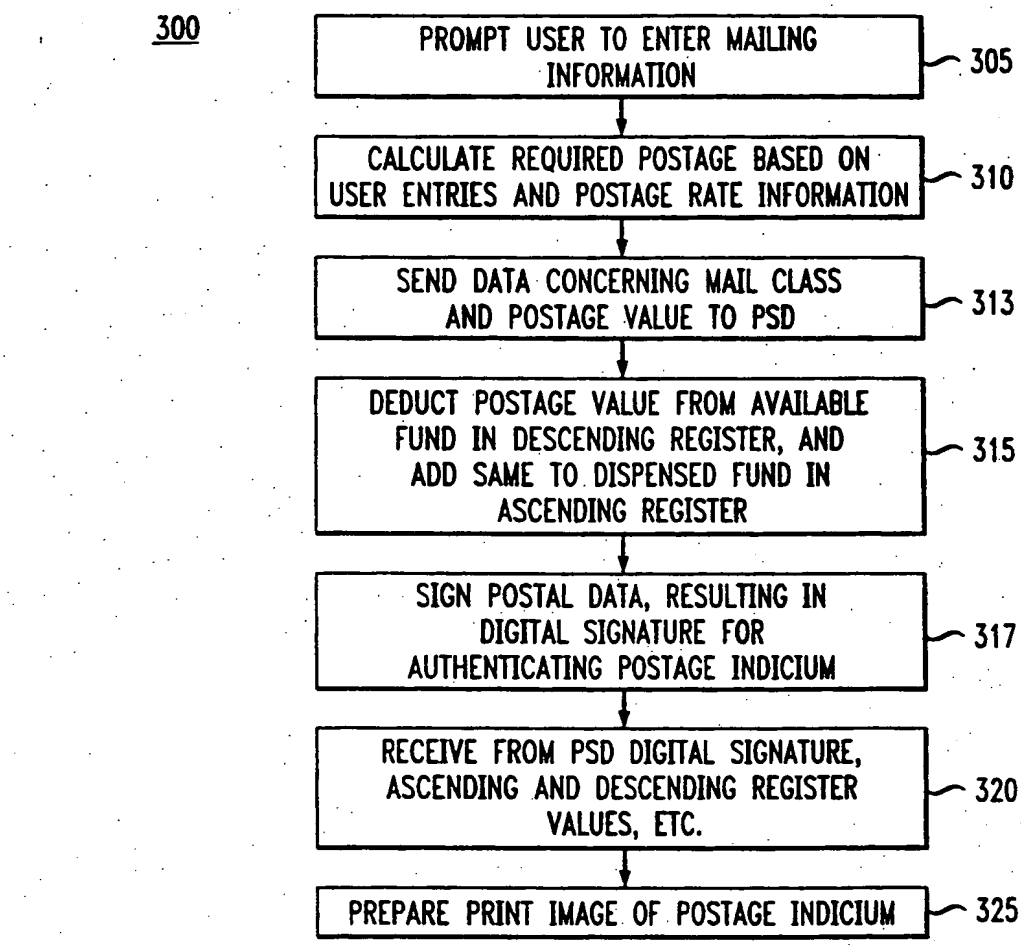


FIG. 5

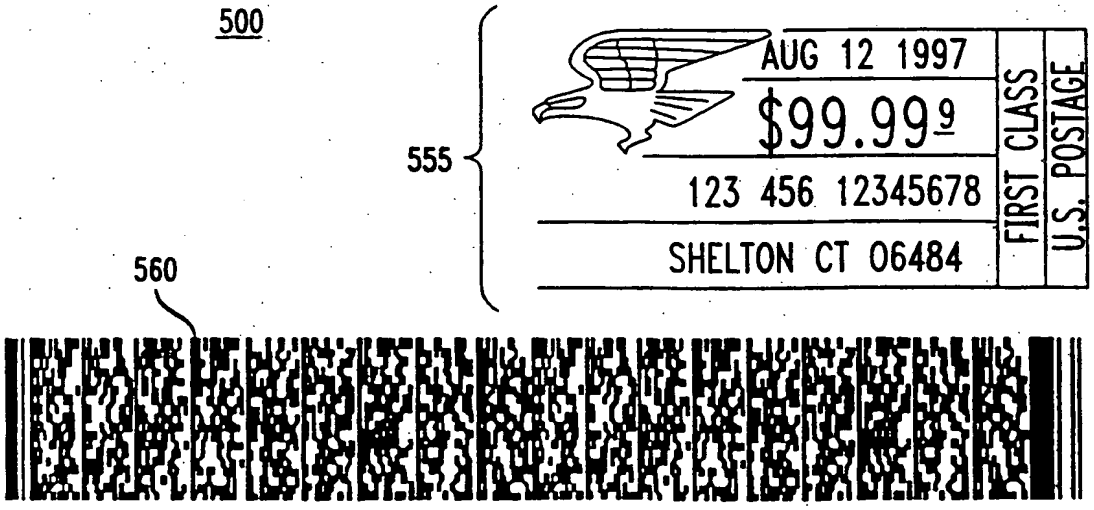


FIG. 6

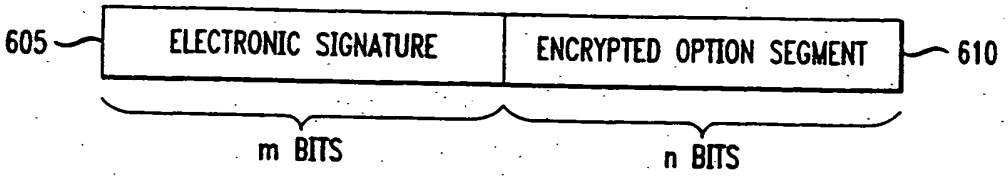


FIG. 7

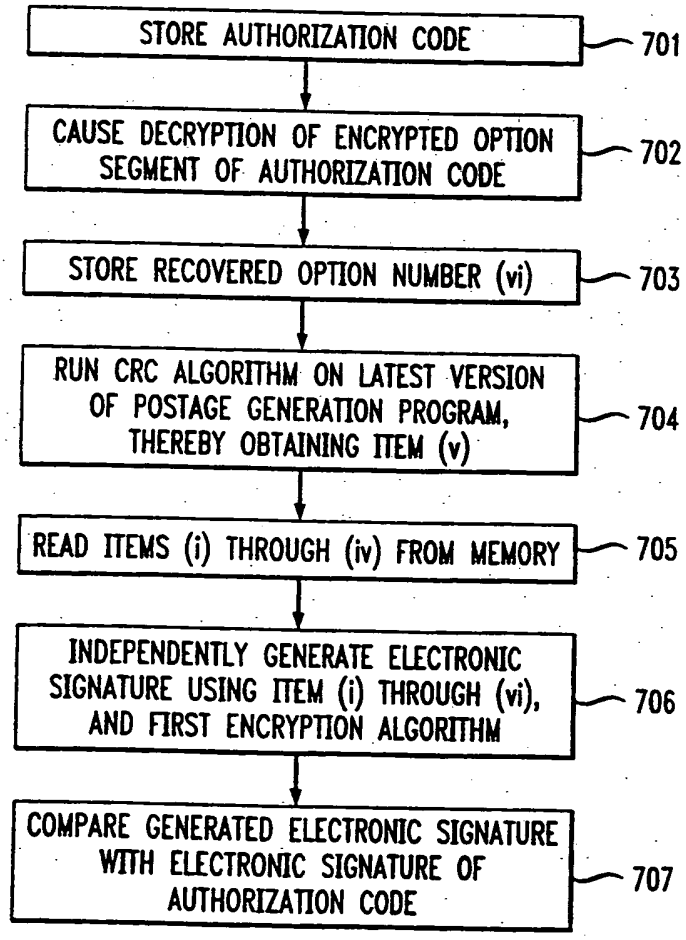
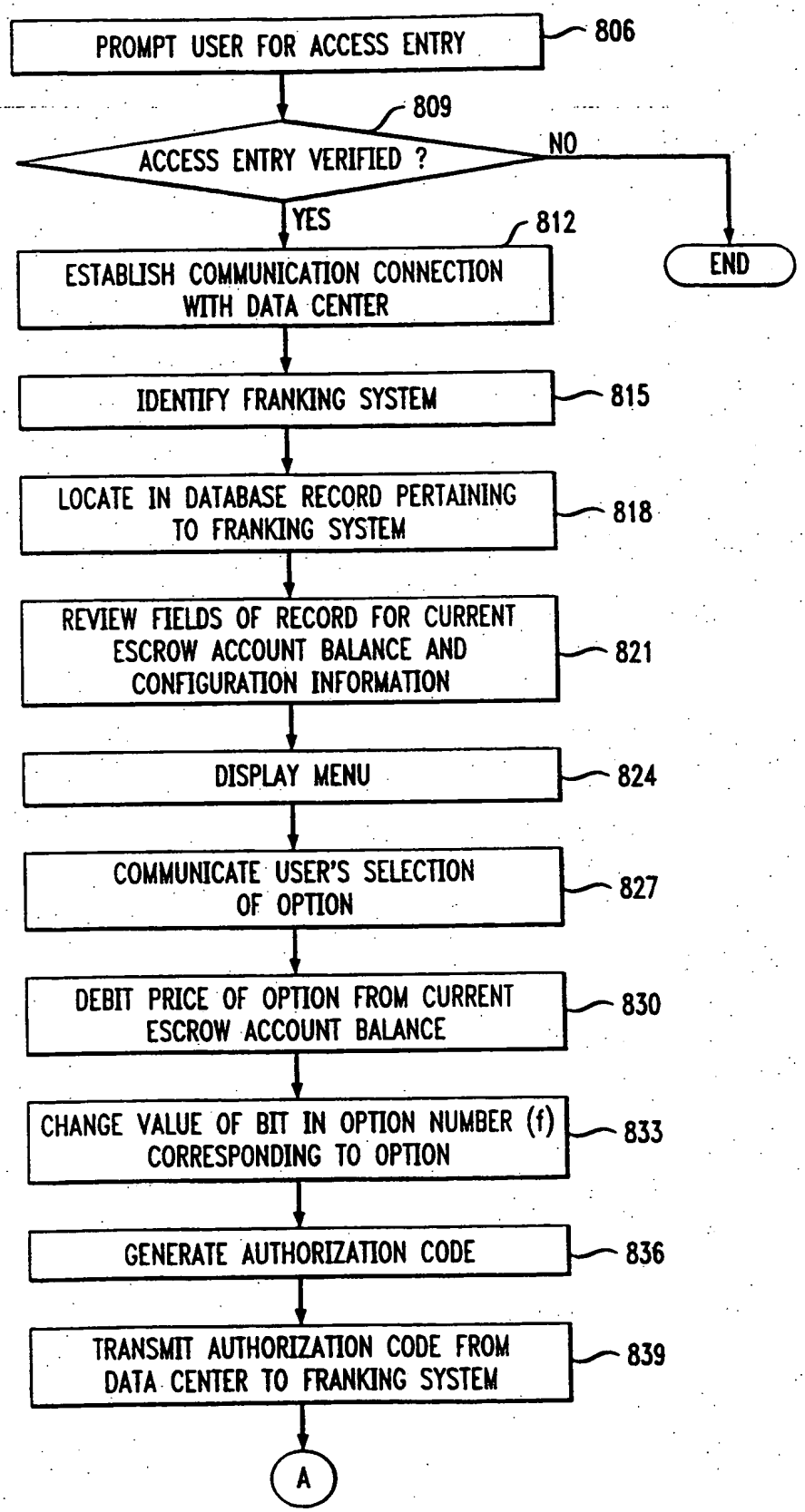


FIG. 8A



TO STEP 841 (FIG. 8B)

FIG. 8B

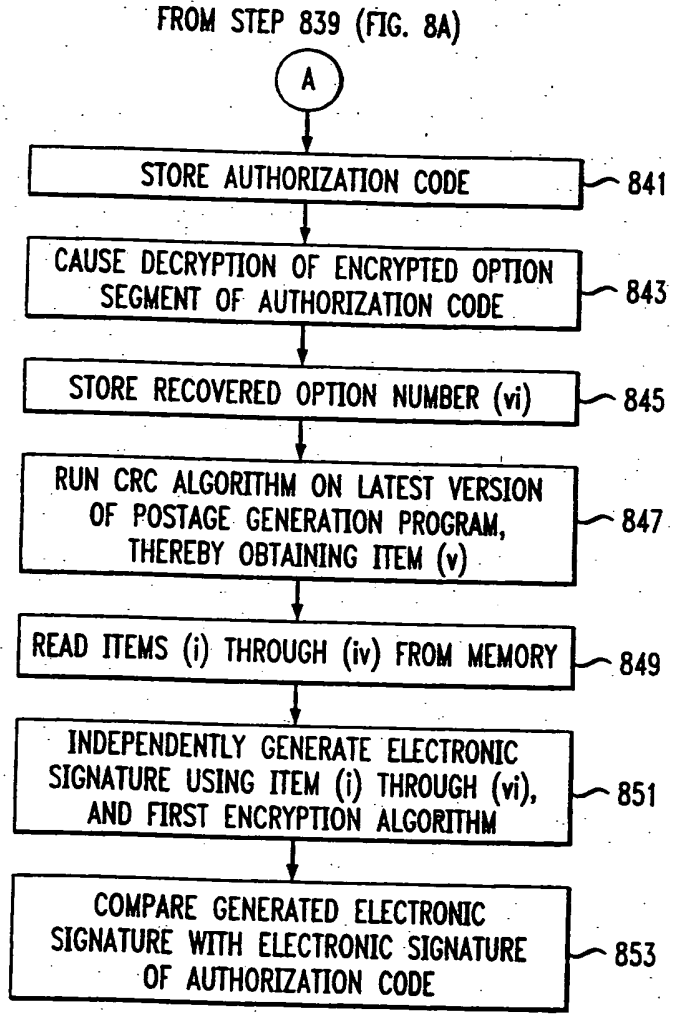


FIG. 9

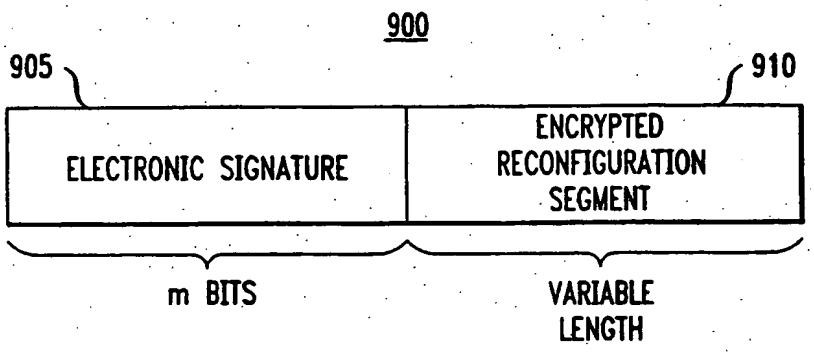


FIG. 10

	MEMORY ADDRESS	MEMORY CONTENT
FEATURE OPTION A	1A2B	0
	1A2C	1
FEATURE OPTION B	1A2D	0
	1A2E	1
FEATURE OPTION C	1A2F	0
	1A30	1

FIG. 11

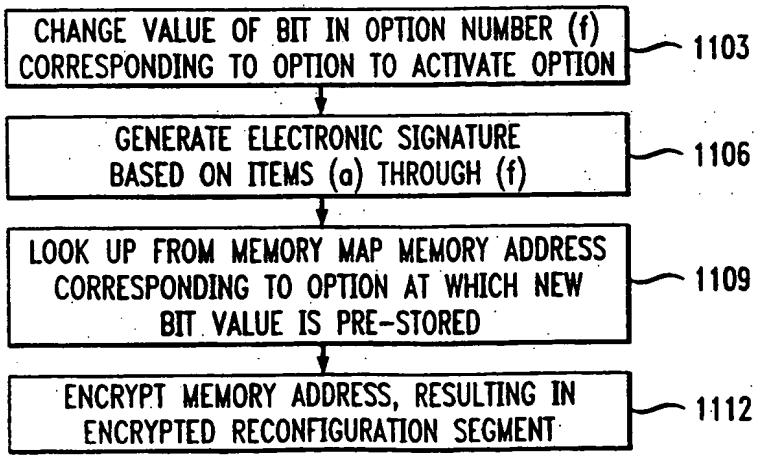


FIG. 12

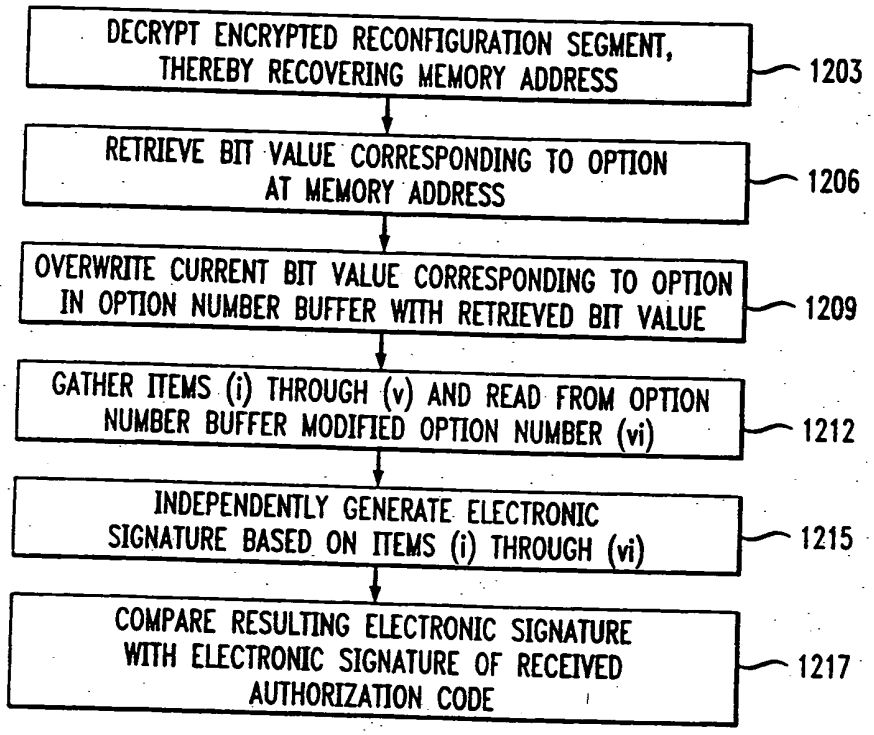


FIG. 13

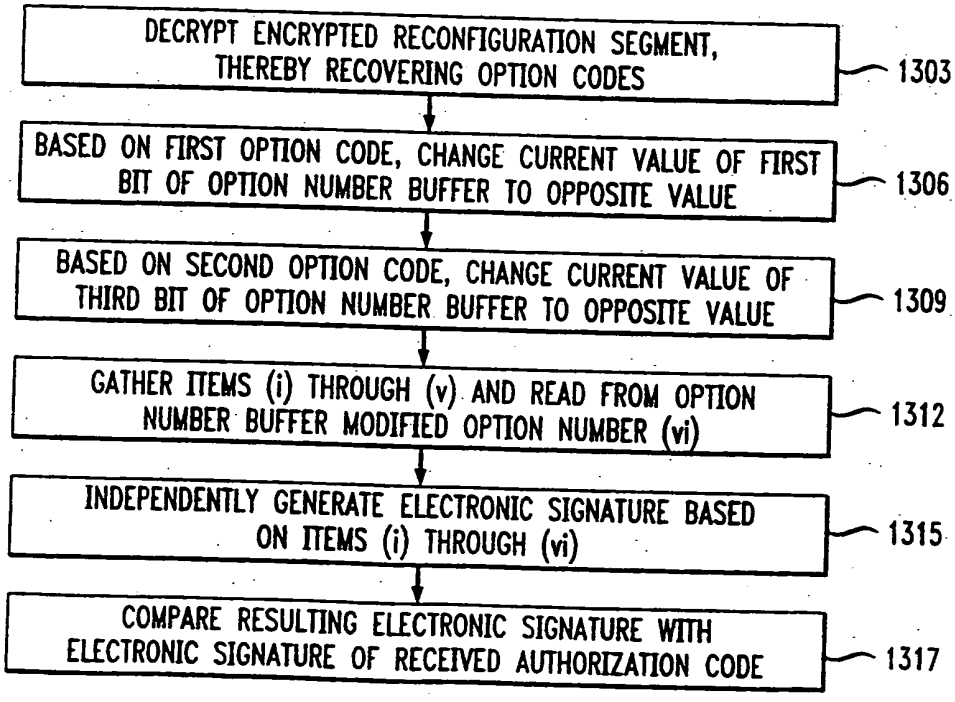


FIG. 14

MEMORY ADDRESS	SOFTWARE VERSION NUMBER
1B12	1
1B13	2
1B14	3
⋮	⋮

FIG. 15

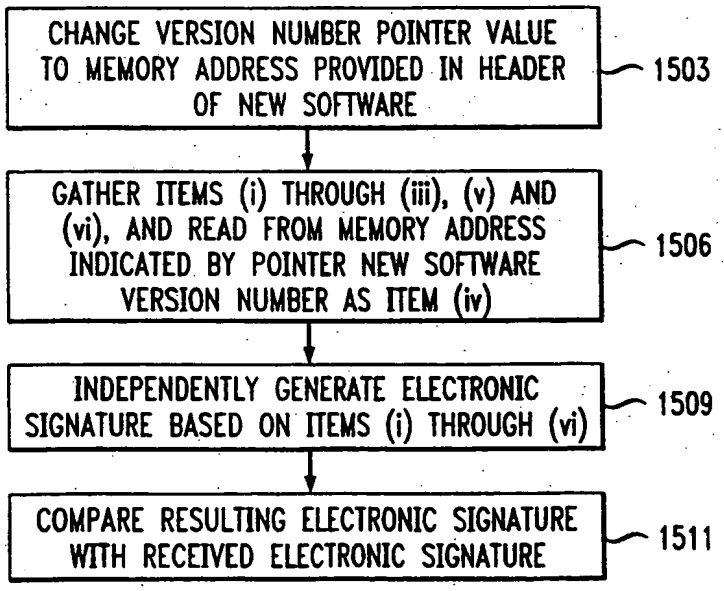


FIG. 16A

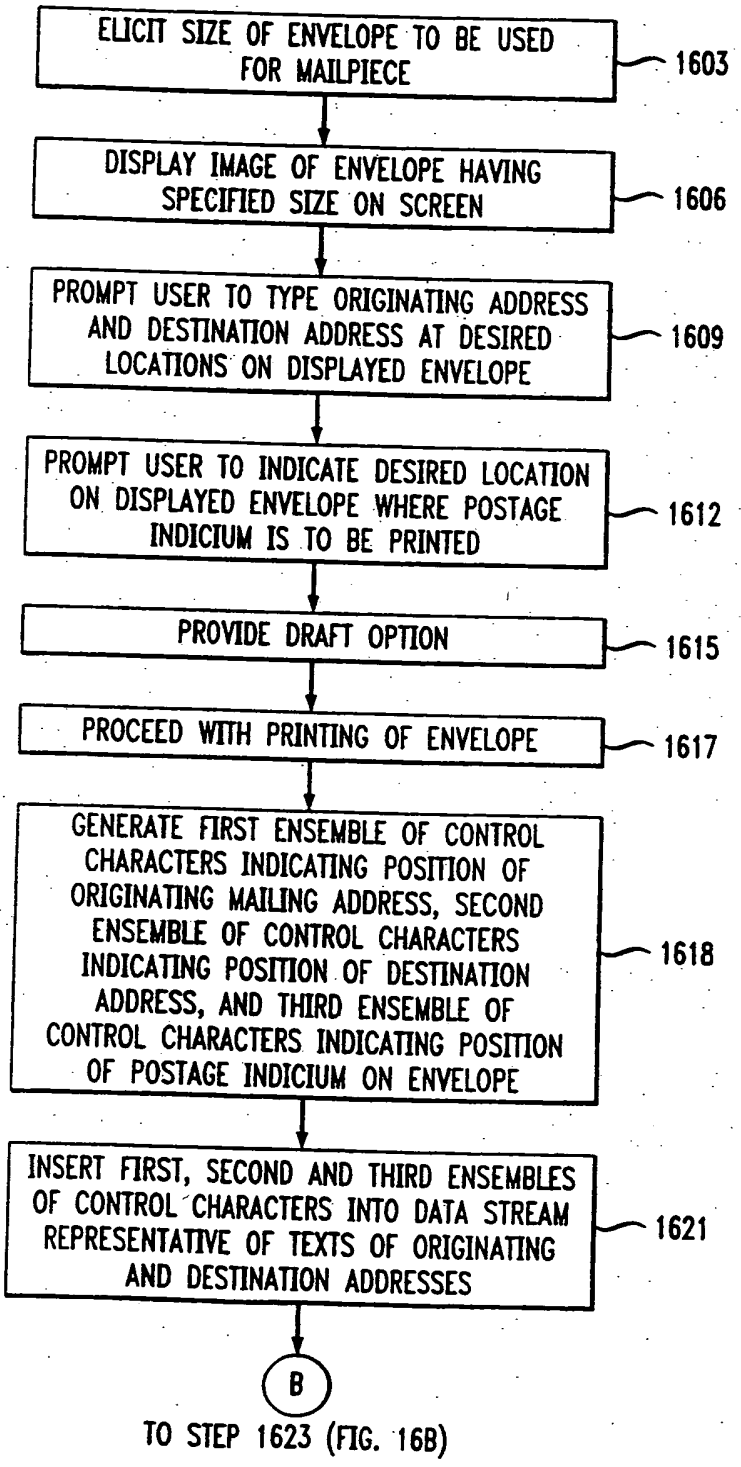


FIG. 16B

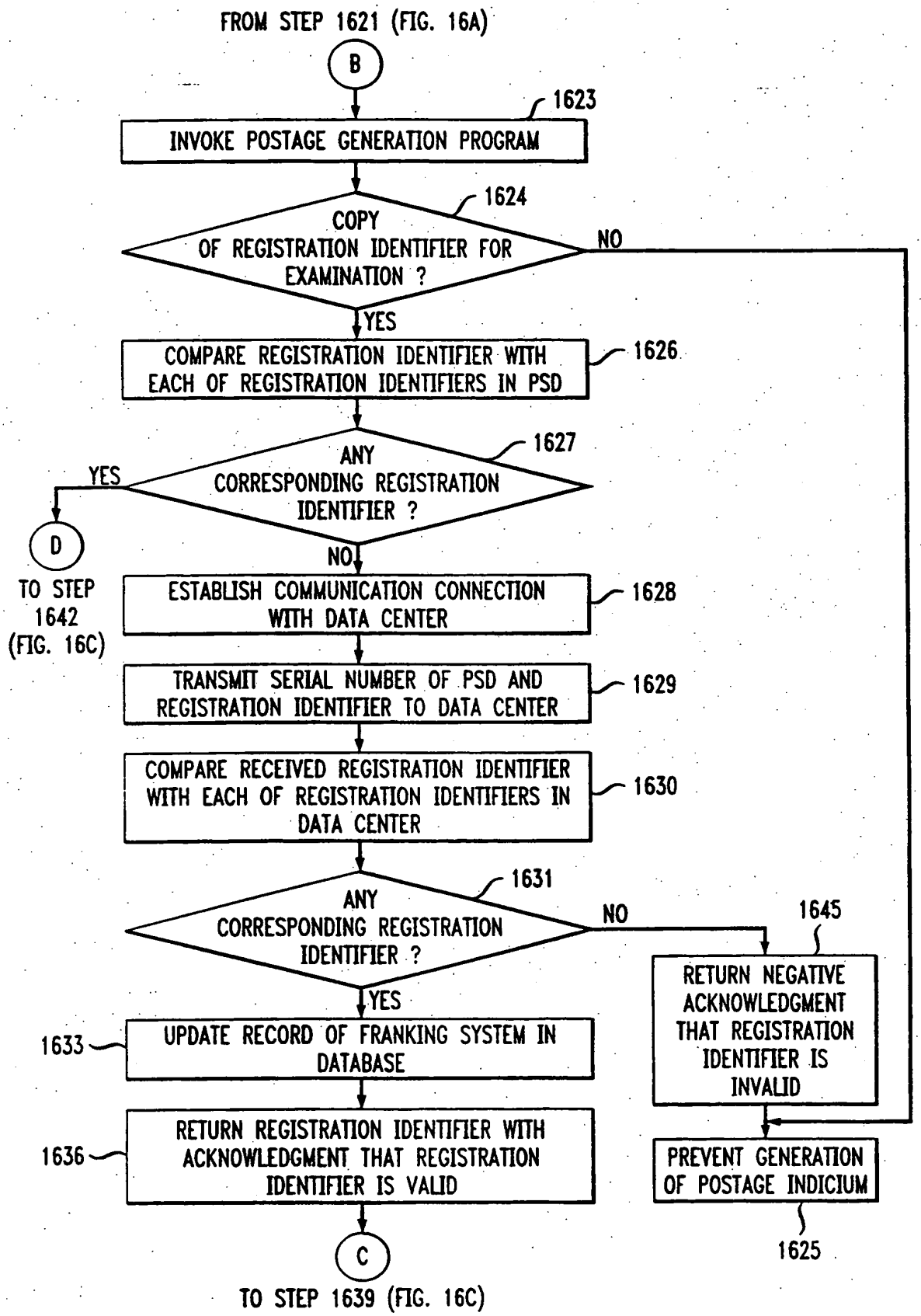
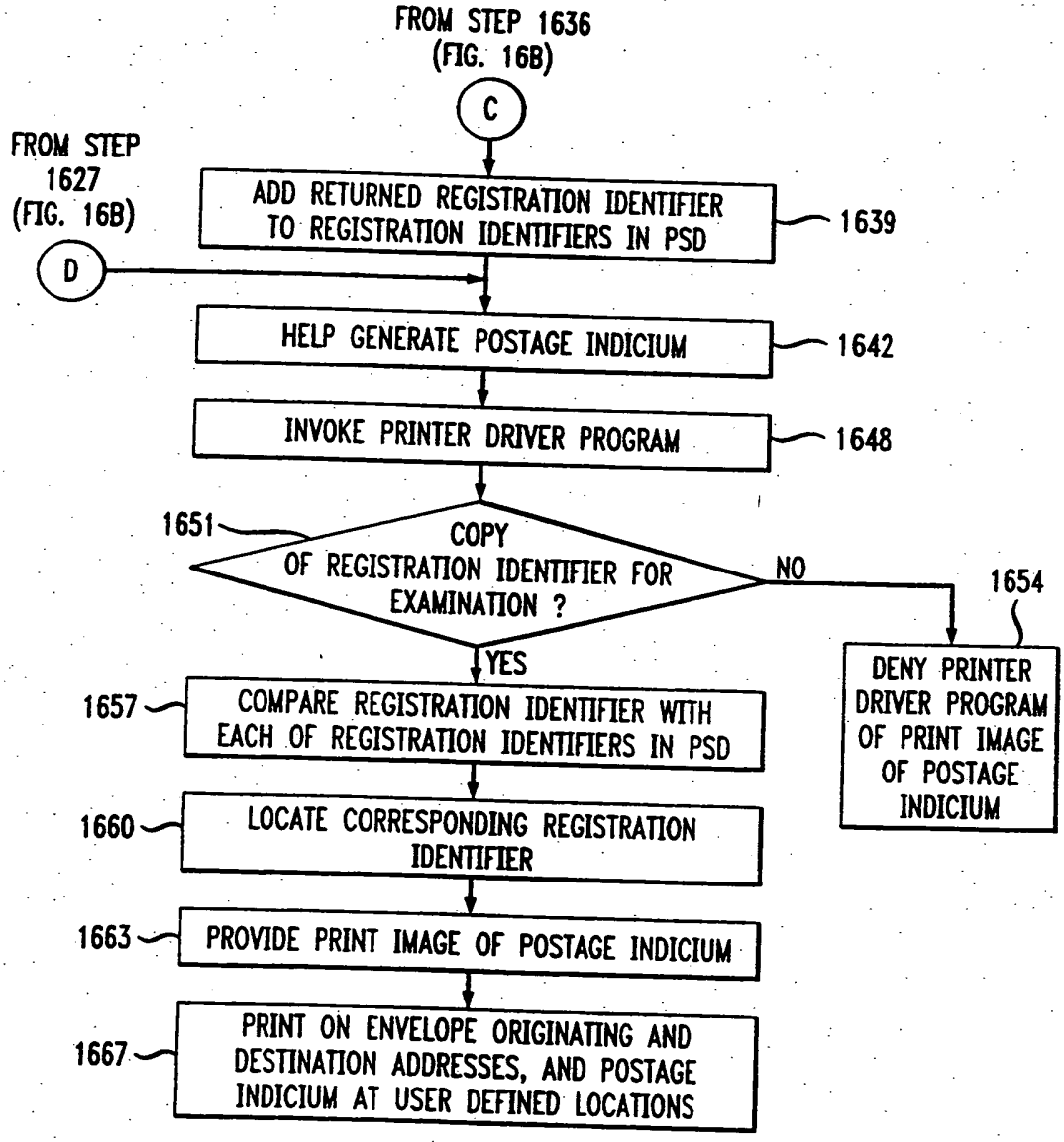


FIG. 16C



INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/13488

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) : G06F 17/00
US CL :705/401

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 705/401

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 4,802,218 A (WRIGHT et al) 31 January 1989, Fig. 8, entire document	1-63
Y,P	US 5,852,813 A (GUENTHER et al) 22 December 1998, entire document	1-63
Y	US 5,680,463 A (WINDEL et al) 21 October 1997, entire document	1-63

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 12 AUGUST 1999	Date of mailing of the international search report 05 OCT 1999
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer For E. TODD VOELTZ Telephone No. (703) 308-3900

Form PCT/ISA/210 (second sheet)(July 1992) ★

CORRECTED VERSION*

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : G06F 17/00 | A1

(11) International Publication Number: WO 99/66422

(43) International Publication Date: 23 December 1999 (23.12.99)

(21) International Application Number: PCT/US99/13488

(22) International Filing Date: 15 June 1999 (15.06.99)

(30) Priority Data:
60/089,212 15 June 1998 (15.06.98) US

(63) Related by Continuation (CON) or Continuation-in-Part (CIP) to Earlier Application
US 08/485,269 (CIP)
Filed on 7 June 1995 (07.06.95)

(71) Applicant *(for all designated States except US)*: ASCOM HASLER MAILING SYSTEMS, INC. [US/US]; 19 Forest Parkway, P.O. Box 858, Shelton, CT 06484-0904 (US).

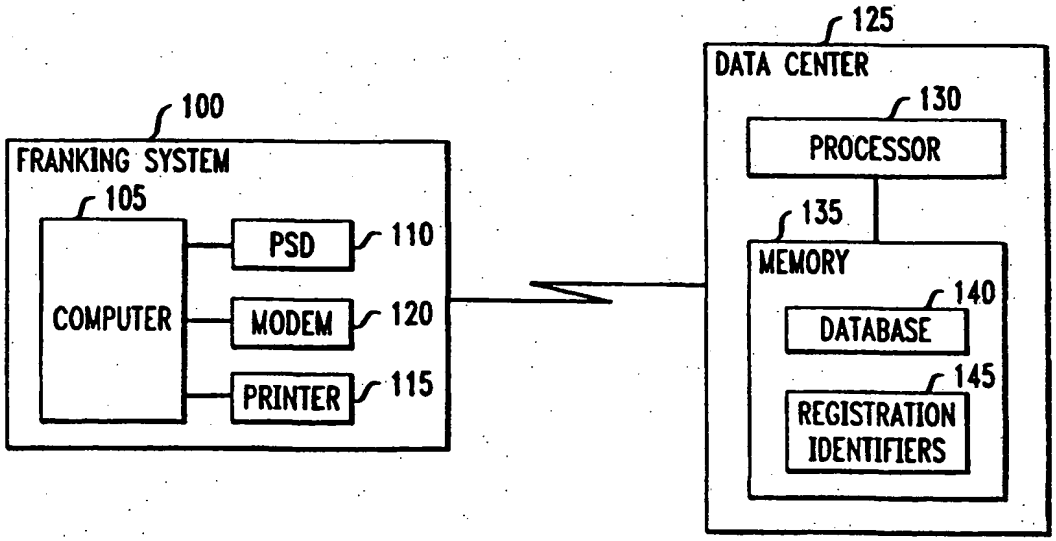
(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: SCHWARTZ, Robert, G. [US/US]; 191 Linden Avenue, Branford, CT 06405 (US). BROOKNER, George, M. [US/US]; 11 Surrey Drive, Norwalk, CT 06851 (US). ESKANDARI, Fetneh [IR/US]; 144 Dove Lane, Middletown, CT 06457 (US). CROWE, Allen, A. [US/US]; 76 Klein Drive, Prospect, CT 06712 (US). SIMCIK, Mark, E. [US/US]; 141 Park Avenue, Bloomfield, CT 06002 (US).

(74) Agent: YIP, Alex, L.; Londa & Traub LLP, 37th floor, 20 Exchange Place, New York, NY 10005 (US).

(81) Designated States: CA, JP, US, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.

(54) Title: TECHNIQUE FOR SECURING A SYSTEM CONFIGURATION OF A POSTAGE FRANKING SYSTEM



(57) Abstract

In a franking system a postal security device (PSD) tracks a postage fund for dispensing postal indicia and enforce the configuration of the franking system. An authorization code, which is particular to the system, is used to verify the system configuration. An unauthorized change in the system configuration causes invalidation of the code and generation of the postal indicia is denied. Data center (125) records configuration information of each franking system (100). The data center generates a valid authorization code for verification in the franking system based on new configuration information. Components added to the system must be preapproved to prevent fraudulent generation of postage indicia. A registration identifier is assigned to each preapproved component which is necessary for interaction with the franking system.

*(Referred to in PCT Gazette No. 09/2000, Section II)

BNSDOCID: <WO___9966422A1_IA>

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania
AM	Armenia
AT	Austria
AU	Australia
AZ	Azerbaijan
BA	Bosnia and Herzegovina
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CU	Cuba
CZ	Czech Republic
DE	Germany
DK	Denmark
EE	Estonia
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GH	Ghana
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IL	Israel
IS	Iceland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgyzstan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakstan
LC	Saint Lucia
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LS	Lesotho
LT	Lithuania
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
MK	The former Yugoslav Republic of Macedonia
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
MX	Mexico
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SG	Singapore
SI	Slovenia
SK	Slovakia
SN	Senegal
SZ	Swaziland
TD	Chad
TG	Togo
TJ	Tajikistan
TM	Turkmenistan
TR	Turkey
TT	Trinidad and Tobago
UA	Ukraine
UG	Uganda
US	United States of America
UZ	Uzbekistan
VN	Viet Nam
YU	Yugoslavia
ZW	Zimbabwe